

Amendments to the Claims

1 Claim 1 (currently amended): A method of improving installation of software packages,
2 comprising steps of:

3 defining an object model as a framework for creating representing a plurality of
4 ~~components of a software installation package and packages including~~ one or more topology
5 objects, wherein the model is independent of any particular software installation package to be
6 created from the model and specifies that each particular software installation package has a suite
7 level and a component level, wherein the suite level serves as a container for one or more
8 components to be included at the component level and each component comprises a plurality of
9 objects and wherein each topology object identifies one or more selected ones of the components;
10 populating the object model to describe a particular software installation package and one
11 or more topologies for deployment of that particular software installation package; and
12 defining one or more rules for execution by a rules engine, wherein each rule specifies
13 one or more conditions and at least one action to be taken when the specified conditions are
14 matched during the execution by the rules engine, and wherein the specified conditions pertain to
15 a target run-time environment and the at least one action ~~may be~~ is used to select from among the
16 topologies.

1 Claim 2 (currently amended): The method according to Claim 1, further comprising the step of
2 instantiating a plurality of objects for the component level and an object for the suite level
3 according to the defined object model, and wherein the populating step populates the instantiated
4 objects.

Serial No. 09/930,325

-5-

RSW920010066US1

1 Claim 3 (original): The method according to Claim 2, wherein the instantiated objects are
2 JavaBeans.

1 Claim 4 (original): The method according to Claim 2, wherein the instantiating step instantiates
2 an object for the particular software installation package and one or more component objects for
3 each software component included in the particular software installation package.

1 Claim 5 (original): The method according to Claim 1, further comprising the steps of:
2 dynamically discovering information pertaining to the target run-time environment;
3 using the dynamically discovered information as input to the execution by the rules
4 engine, wherein the execution results in matching a selected one of the rules;
5 automatically selecting, based upon the at least one action in the matching rule, at least
6 one of the topologies for deployment; and
7 using the populated object model to install the particular software installation package
8 using the selected topology.

1 Claim 6 (original): The method according to Claim 5, wherein the step of using the populated
2 object model further comprises the steps of:
3 identifying one or more target machines on which the particular software installation
4 package is to be installed;
5 downloading the particular software installation package to the identified target

Serial No. 09/930,325

-6-

RSW920010066US1

6 machines; and

7 performing an installation at each of the identified target machines using the downloaded
8 particular software installation package.

1 Claim 7 (original): The method according to Claim 6, further comprising the step of
2 authenticating a server on which the downloading step operates prior to an operation of the step
3 of performing the installation.

1 Claim 8 (original): The method according to Claim 5, wherein the step of using the dynamically
2 discovered information as input to the execution by the rules engine also serves to configure one
3 or more values needed by the selected topology.

1 Claim 9 (currently amended): A system for improving installation of software packages,
2 comprising:

3 means for defining an object model as a framework for creating representing a plurality of
4 components of a software installation package and packages including one or more topology
5 objects, wherein the model is independent of any particular software installation package to be
6 created from the model and specifies that each particular software installation package has a suite
7 level and a component level, wherein the suite level serves as a container for one or more
8 components to be included at the component level and each component comprises a plurality of
9 objects and wherein each topology object identifies one or more selected ones of the components;
10 means for populating the object model to describe a particular software installation

Serial No. 09/930,325

-7-

RSW920010066US1

11 package and one or more topologies for deployment of that particular software installation
12 package; and
13 means for defining one or more rules for execution by a rules engine, wherein each rule
14 specifies one or more conditions and at least one action to be taken when the specified conditions
15 are matched during the execution by the rules engine, and wherein the specified conditions
16 pertain to a target run-time environment and the at least one action may be is used to select from
17 among the topologies.

1 Claim 10 (original): The system according to Claim 9, further comprising:
2 means for dynamically discovering information pertaining to the target run-time
3 environment;
4 means for using the dynamically discovered information as input to the execution by the
5 rules engine, wherein the execution results in matching a selected one of the rules;
6 means for automatically selecting, based upon the at least one action in the matching rule,
7 at least one of the topologies for deployment; and
8 means for using the populated object model to install the particular software installation
9 package using the selected topology.

1 Claim 11 (original): The system according to Claim 10, wherein the means for using the
2 populated object model further comprises:
3 means for identifying one or more target machines on which the particular software
4 installation package is to be installed;

means for downloading the particular software installation package to the identified target machines; and

means for performing an installation at each of the identified target machines using the downloaded particular software installation package.

Claim 12 (original): The system according to Claim 10, wherein the means for using the dynamically discovered information as input to the execution by the rules engine also serves to configure one or more values needed by the selected topology.

Claim 13 (currently amended): A computer program product for improving installation of software packages, the computer program product embodied on one or more computer-readable media and comprising:

computer-readable program code means for defining an object model as a framework for creating representing a plurality of components of a software installation package and packages including one or more topology objects, wherein the model is independent of any particular software installation package to be created from the model and specifies that each particular software installation package has a suite level and a component level, wherein the suite level serves as a container for one or more components to be included at the component level and each component comprises a plurality of objects and wherein each topology object identifies one or more selected ones of the components;

computer-readable program code means for populating the object model to describe a particular software installation package and one or more topologies for deployment of that

14 particular software installation package; and
15 computer-readable program code means for defining one or more rules for execution by a
16 rules engine, wherein each rule specifies one or more conditions and at least one action to be
17 taken when the specified conditions are matched during the execution by the rules engine, and
18 wherein the specified conditions pertain to a target run-time environment and the at least one
19 action ~~may be~~ is used to select from among the topologies.

1 Claim 14 (original): The computer program product according to Claim 13, further comprising:
2 computer-readable program code means for dynamically discovering information
3 pertaining to the target run-time environment;
4 computer-readable program code means for using the dynamically discovered information
5 as input to the execution by the rules engine, wherein the execution results in matching a selected
6 one of the rules;
7 computer-readable program code means for automatically selecting, based upon the at
8 least one action in the matching rule, at least one of the topologies for deployment; and
9 computer-readable program code means for using the populated object model to install
10 the particular software installation package using the selected topology.

1 Claim 15 (original): The computer program product according to Claim 14, wherein the
2 computer-readable program code means for using the populated object model further comprises:
3 computer-readable program code means for identifying one or more target machines on
4 which the particular software installation package is to be installed;

5 computer-readable program code means for downloading the particular software
6 installation package to the identified target machines; and
7 computer-readable program code means for performing an installation at each of the
8 identified target machines using the downloaded particular software installation package.

1 Claim 16 (original): The computer program product according to Claim 14, wherein the
2 computer-readable program code means for using the dynamically discovered information as
3 input to the execution by the rules engine also serves to configure one or more values needed by
4 the selected topology.